#DATA:

CREATE DATABASE ORG123;

SHOW DATABASES;

USE ORG123;

CREATE TABLE Worker (

WORKER\_ID INT NOT NULL PRIMARY KEY AUTO\_INCREMENT,

FIRST\_NAME CHAR(25),

LAST\_NAME CHAR(25),

SALARY INT(15),

JOINING\_DATE DATETIME,

DEPARTMENT CHAR(25)

);

INSERT INTO Worker

(WORKER\_ID, FIRST\_NAME, LAST\_NAME, SALARY, JOINING\_DATE, DEPARTMENT) VALUES

(001, 'Monika', 'Arora', 100000, '14-02-20 09.00.00', 'HR'),

(002, 'Niharika', 'Verma', 80000, '14-06-11 09.00.00', 'Admin'),

(003, 'Vishal', 'Singhal', 300000, '14-02-20 09.00.00', 'HR'),

(004, 'Amitabh', 'Singh', 500000, '14-02-20 09.00.00', 'Admin'),

(005, 'Vivek', 'Bhati', 500000, '14-06-11 09.00.00', 'Admin'),

(006, 'Vipul', 'Diwan', 200000, '14-06-11 09.00.00', 'Account'),

(007, 'Satish', 'Kumar', 75000, '14-01-20 09.00.00', 'Account'),

(008, 'Geetika', 'Chauhan', 90000, '14-04-11 09.00.00', 'Admin');

CREATE TABLE Bonus (

WORKER\_REF\_ID INT,

BONUS\_AMOUNT INT(10),

BONUS\_DATE DATETIME,

FOREIGN KEY (WORKER\_REF\_ID)

REFERENCES Worker(WORKER\_ID)

ON DELETE CASCADE

);

INSERT INTO Bonus

(WORKER\_REF\_ID, BONUS\_AMOUNT, BONUS\_DATE) VALUES

(001, 5000, '16-02-20'),

(002, 3000, '16-06-11'),

(003, 4000, '16-02-20'),

(001, 4500, '16-02-20'),

(002, 3500, '16-06-11');

CREATE TABLE Title (

WORKER\_REF\_ID INT,

WORKER\_TITLE CHAR(25),

AFFECTED\_FROM DATETIME,

FOREIGN KEY (WORKER\_REF\_ID)

REFERENCES Worker(WORKER\_ID)

ON DELETE CASCADE

);

INSERT INTO Title

(WORKER\_REF\_ID, WORKER\_TITLE, AFFECTED\_FROM) VALUES

(001, 'Manager', '2016-02-20 00:00:00'),

(002, 'Executive', '2016-06-11 00:00:00'),

(008, 'Executive', '2016-06-11 00:00:00'),

(005, 'Manager', '2016-06-11 00:00:00'),

(004, 'Asst. Manager', '2016-06-11 00:00:00'),

(007, 'Executive', '2016-06-11 00:00:00'),

(006, 'Lead', '2016-06-11 00:00:00'),

(003, 'Lead', '2016-06-11 00:00:00');

#QUE 1

SELECT DISTINCT DEPARTMENT FROM Worker;

# The GROUP BY clause will ensure that non-duplicate values from the DEPARTMENT column are retrieved.

#QUE 2

SELECT \* FROM Worker

ORDER BY FIRST\_NAME ASC, DEPARTMENT DESC;

# I want to see all the details about each worker, so I used SELECT \* which means "show everything" then i used ORDER BY to sort the list in a specific way.

#QUE 3

SELECT \* FROM Worker

WHERE FIRST\_NAME LIKE '%a%';

# i used '%a%' to select only those FIRST\_NAMES that have the letter 'a' within their string.

#QUE 4

SELECT \* FROM Worker

WHERE FIRST\_NAME LIKE '\_\_\_\_\_\_' AND FIRST\_NAME LIKE '%h';

# here 6 underscores will ensure the 6 characters in name and %h will ensure name ends with h

#QUE 5

SELECT \* FROM Worker

WHERE SALARY BETWEEN 100000 AND 500000;

# here i used AND operator to test both the conditions

#QUE 6

SELECT \* FROM Worker

WHERE YEAR(JOINING\_DATE) = 2014 AND MONTH(JOINING\_DATE) = 2;

#here i used the YEAR and MONTH clause to check the joining dates

#QUE 7

SELECT COUNT(\*) FROM Worker

WHERE DEPARTMENT = 'Admin';

# this clause checks for department here.

#QUE 8

SELECT FIRST\_NAME, LAST\_NAME

FROM Worker

WHERE SALARY >= 50000 AND SALARY <= 100000;

# AND clause is used to find the results that lies in between the given values

#QUE 9

SELECT DEPARTMENT, COUNT(WORKER\_ID) AS Worker\_Count

FROM Worker

GROUP BY DEPARTMENT

ORDER BY Worker\_Count DESC;

# it will count workers in each group and group the department .

#QUE 10

SELECT \* FROM Worker

WHERE WORKER\_ID IN (

SELECT MANAGER\_ID FROM Worker

);

# it will combine rows of worker where id will match.

#QUE 11

)SELECT MIN(SALARY) AS Second\_Lowest\_Salary

FROM Worker

WHERE SALARY > (

SELECT MIN(SALARY) FROM Worker

);

# the inner sub query will find the lowest salary in the worker table.

#QUE 12

SELECT \*

FROM Worker

WHERE SALARY IN (

SELECT SALARY

FROM Worker

GROUP BY SALARY

HAVING COUNT(\*) > 1

);

# the inner clause will find all salary value which occured more than once in worker table.

#QUE 13

SELECT MAX(SALARY) AS Second\_Highest\_Salary

FROM Worker

WHERE SALARY < (SELECT MAX(SALARY) FROM Worker);

# it finds the highest salary

#QUE 14

SELECT \* FROM Worker

WHERE WORKER\_ID = 1

UNION ALL

SELECT \* FROM Worker

WHERE WORKER\_ID = 1;

# This query selects the same row twice using UNION ALL . It keeps duplicates.

#QUE 15

# i will calculate the total number of rows then divide it by two , and then use that number with LIMIT.

#QUE 16

SELECT department

FROM Worker

GROUP BY department

HAVING COUNT(\*) < 3;

#it will include departments where the calculated count of workers is less than 3.

#QUE 17

SELECT DEPARTMENT, COUNT(WORKER\_ID) AS total\_people

FROM Worker

GROUP BY DEPARTMENT;

# it will select name of each department and count the number of worker ids.

#QUE 18

SELECT \* FROM Worker

ORDER BY WORKER\_ID ASC

LIMIT 5;

# it will get only 1st 5 rows from the sorted ans

#QUE 19

I think his query shows the name, department, and salary of employees who get the highest salary in their department and checks for each employee if their salary is the highest in their department by using a subquery with MAX(salary).

#QUES 20

SELECT DISTINCT salary

FROM employees

ORDER BY salary DESC

LIMIT 3;

# i think here it will show top 3 diff highest salaries from the table .

#QUES 21

SELECT FIRST\_NAME, LAST\_NAME, DEPARTMENT, SALARY

FROM Worker

WHERE DEPARTMENT IN ('Account', 'Admin')

AND SALARY = (

SELECT MIN(SALARY) #selects the lowest salary where department is account and admin

FROM Worker

WHERE DEPARTMENT IN ('Account', 'Admin')

);